

Amendments to the Claims:

This listing of claims replaces all prior versions and listings of claims in the application:

Listing of Claims:

1. (currently amended): A method for streaming a multimedia document, the method comprising:

receiving a stream including a file that integrates media content with choreography information within each of at least two objects of the file, each of the objects including media content data and choreography information associated therewith, the choreography information comprising data indicating an author-designated relationship between the objects of the file that defines [[a]] an author-designated temporal order of presentation between the objects; and

before all objects of the file are received, beginning to render media content encapsulated within the file based on the choreography information associated with objects received so as to enable ~~an ordered~~ display of the objects received based on the temporal order defined by the choreography information, wherein the temporal order is maintained independent of a recipient or a web server and ~~unaffected by an input of a recipient, the ordered display being~~ independent of a bandwidth of a communications channel used to send the multimedia document.

2. (previously presented): The method of claim 1, further comprising changing at least one of the objects in the file.

3. (previously presented): The method of claim 1, further comprising adding at least one object to the file.

4. (previously presented): The method of claim 1 wherein the file is displayed in a window on a computer display, the method further comprising:
creating an exclusionary area within the window; and
locating an object within the exclusionary area, the object being selected from a group of objects including a framed image, a slide show, framed text, sound data, a separator, or a hyperlink.

5. (previously presented): The method of claim 1 wherein the file includes splash image data defining a splash image, the method further comprising locating the splash image data within the file such that the splash image is displayed on a computer display as the splash image data is received by a receiver coupled to the computer display.

6. (previously presented): The method of claim 5, further comprising locating update splash data that further defines the splash image within the file such that the splash image is updated on the computer display as the receiver receives the update splash data.

7. (original): The method of claim 1, further comprising providing each object with an address indicating a player that plays the object.

8. (previously presented): The method of claim 1, further comprising compressing the data for the object in at least one object.

9. (previously presented): The method of claim 1 further comprising:
creating an unknown object in the file; and
locating player data within the unknown object defining a player that plays the unknown object.

10. (currently amended): A computer system having a memory storing a file structure, the file structure comprising:

at least two objects that integrate media content with choreography information within each of at least two objects of a file, each of the objects including media content data and choreography information associated therewith, the choreography information comprising data indicating an author-designated relationship between the objects of the file that defines [[a]] an author-designated temporal order of presentation between the objects; and

the file structure enabling the beginning of rendering, before all objects of the file are received, of the media content encapsulated within the file according to the choreography information associated with objects received so as to enable ~~an ordered~~ display of the objects received based on the temporal order defined by the choreography information, wherein the temporal order is maintained independent of a recipient or a web server and unaffected by an input of the recipient, and the ordered display being independent of a bandwidth of a communications channel used to send the multimedia document.

11. (previously presented): The computer system of claim 10 wherein at least one object comprises one of a textual file format, an image file format, and a sound file format.

12. (cancelled).

13. (previously presented): The computer system of claim 10 wherein two or more objects have at least one common attribute, including at least one of a command for perception of the object, an ability to pass and receive a message, and an ability to supply and retrieve the data embodied in the object.

14. (previously presented): The computer system of claim 10 wherein at least one object is a generic element of a hierarchical data file structure, such that any combination of objects can be grouped together to form a part of the multimedia document.

15. (previously presented): The computer system of claim 10 wherein the document forms a code segment that receives image information; and wherein the image information is used to construct an image frame for a framed image that is part of the multimedia document.

16. (previously presented): The computer system of claim 15 wherein the framed image has an image data format; and wherein a decoder determines the image data format and encapsulates the framed image with the image frame.

17-30. (cancelled).

31. (previously presented) The method of claim 1, wherein the choreography information further comprises:

a header;

an object archive for storing information about one or more objects, the object archive including information about the relationship of the object file with the document; and

a multiplex section including data for the objects in the document.

32. (previously presented): The method of claim 31, wherein the objects in the multiplex section are each played by a player as the multiplexed object is received by a receiver.

33. (previously presented): The method of claim 31, wherein the data for the objects is interleaved in the multiplex section.

34. (previously presented): The method of claim 31, wherein the object archive includes data defining a geometry of the document.

35. (previously presented): The method of claim 31, wherein one or more objects is defined by at least one data slice; and wherein the multiplex section further includes:

- an object number counter indicating the number of objects;
- a plurality of object descriptions, each object description describing a corresponding one of the objects; and
- a choreography group providing information about a first group of objects.

36. (previously presented): The method of claim 35, wherein the choreography group further comprises:

- a group object counter indicating the number of objects in the choreography group;
- size and type data for each object;
- header data; and
- the data slices of the objects interleaved together.

37. (previously presented): The method of claim 35, wherein the choreography group includes data slices of the objects interleaved in a predetermined manner.

38. (previously presented): The method of claim 35, further comprising providing a first player pointer including an address of a player that plays the choreography group.

39. (previously presented): The method of claim 35, further comprising placing one or more slice size data blocks before one or more of the interleaved data slices, each slice size data block corresponding to a data slice and providing a size of the corresponding data slice.

40. (previously presented): The method of claim 31, further comprising a non-multiplex section following the multiplex section, the non-multiplex section including one or more separate objects that are not played by a player as the separate object files are received by a receiver.

41. (previously presented): The computer system of claim 10, wherein the choreography information further comprises:

a header;

an object archive for storing information about one or more objects, the object archive including information about the relationship of the object file with the document; and

a multiplex section including data for the objects in the document.

42. (previously presented): The computer system of claim 41, wherein the objects in the multiplex section are each played by a player as the multiplexed object is received by a receiver.

43. (previously presented): The computer system of claim 41, wherein the data for the objects is interleaved in the multiplex section.

44. (previously presented): The computer system of claim 41, wherein the object archive includes data defining a geometry of the document.

45. (previously presented): The computer system of claim 41, wherein one or more objects is defined by at least one data slice, and wherein the multiplex section further includes:

an object number counter indicating the number of objects;

a plurality of object descriptions, each object description describing a corresponding one of the objects; and

a choreography group providing information about a first group of objects.

46. (previously presented): The computer system of claim 45, wherein the choreography group further comprises:

a group object counter indicating the number of objects in the choreography group;

size and type data for each object;

header data; and

the data slices of the objects interleaved together.

47. (previously presented): The computer system of claim 45, wherein the choreography group includes data slices of the objects interleaved in a predetermined manner.

48. (previously presented): The computer system of claim 45, further comprising providing a first player pointer including an address of a player that plays the choreography group.

49. (previously presented): The computer system of claim 45, further comprising placing one or more slice size data blocks before one or more of the interleaved data slices, each slice size data block corresponding to a data slice and providing a size of the corresponding data slice.

50. (previously presented): The computer system of claim 41, further comprising a non-multiplex section following the multiplex section, the non-multiplex section including one or more separate objects that are not played by a player as the separate object files are received by a receiver.

51-62. (cancelled).

63. (currently amended): The method of claim 1 in which the ~~ordered~~ display is independent of a recipient software program used to render the objects.

64. (currently amended): The method of claim 63 wherein the recipient software comprises a browser, and wherein the ~~ordered~~ display is independent of the browser.

65. (currently amended): The computer system of claim 10 in which the ~~ordered~~ display is independent of a recipient software program used to render the objects.

66. (currently amended): The computer system of claim 65 wherein the recipient software comprises a browser, and wherein the ~~ordered~~ display is independent of the browser.

67 – 99. (cancelled).

100. (currently amended): A method for producing a streaming document, the method comprising:

receiving specification of media content by an author;

receiving designation by the author of choreography information that indicates at least an intended order of presentation for the specified media content;

generating a single file that integrates the media content with the choreography information, wherein generating the single file comprises encapsulating within the single file at least two objects, each object including media content data and choreography information associated therewith, the choreography information comprising data defining an author-designated relationship between the objects of the single file that defines [[a]] an author-designated temporal order of presentation between the objects; and

before all objects of the file are received by a recipient, enabling the recipient to begin rendering the media content encapsulated within the file according to the temporal order defined by the choreography information associated with objects received, wherein the temporal order is maintained independent of a recipient or a web server.

101. (new): The method of claim 100 in which the temporal order is independent of a recipient input.

102. (new): The method of claim 100 in which the temporal order is independent of a recipient hardware configuration.

103. (new): The method of claim 100 in which the temporal order is independent of a recipient software configuration.

104. (new): The method of claim 1 in which the temporal order is independent of a recipient input.

105. (new): The method of claim 1 in which the temporal order is independent of a recipient hardware configuration.

106. (new): The method of claim 1 in which the temporal order is independent of a recipient software configuration.

107. (new): The computer system of claim 10 in which the temporal order is independent of a recipient input.

108. (new): The computer system of claim 10 in which the temporal order is independent of a recipient hardware configuration.

109. (new): The computer system of claim 10 in which the temporal order is independent of a recipient software configuration.